

Claims

1  
2  
3 1. A computer-implemented method for detecting a macro virus  
4 in code adapted for use on a digital computer, said method  
5 comprising the steps of:

6 analyzing the code to determine whether said code

7  
8 contains instructions causing a macro to be moved to  
9 a global environment;

10 when the code contains instructions causing a macro to be  
11 moved to a global environment, flagging said macro;

12 analyzing the code to determine whether said code

13  
14 contains instructions causing the flagged macro to be  
15 copied to a local document; and

16 when the code contains instructions causing the flagged  
17 macro to be copied to a local document, declaring  
18 that said flagged macro contains a macro virus.  
19

20 2. The method of claim 1 wherein the macro is contained  
21 within a module.

22 3. The method of claim 1 wherein the code is associated with a  
23 Microsoft Excel spreadsheet application.

24 4. The method of claim 1 wherein the code is written in the  
25 Visual Basic language.  
26

27 5. The method of claim 4 wherein the step of analyzing the  
28 code to determine whether said code contains instructions causing a

1 macro to be moved to a global environment comprises determining  
2 whether a SaveAs command is present in the code.

3  
4 6. The method of claim 4 wherein the step of analyzing the  
5 code to determine whether said code contains instructions causing  
6 the flagged macro to be copied to a local document comprises  
7 determining whether a Copy command is present in the code.

8  
9 7. The method of claim 1 wherein each analyzing step  
10 concatenates strings when said analyzing step encounters a  
11 concatenation operator within the code.

12  
13 8. The method of claim 1 wherein each analyzing step makes  
14 substitutions for variable names when the code contains variable  
15 names that are proxied.

16  
17 9. The method of claim 1 wherein each analyzing step traces  
18 the values of parameter variables when the code contains  
19 instructions that are invoked by other code.

20  
21 10. The method of claim 1 wherein each analyzing step  
22 substitutes object names when the code is written in an object  
23 oriented programming language and when the code contains  
24 substituted object names.

25  
26 11. The method of claim 1 further comprising the step of  
27 deleting the macro virus.

28  
12. The method of claim 1 wherein publicly identified and  
publicly unidentified macro viruses are detected.

1           13. A method for detecting publicly identified and publicly  
2 unidentified macro viruses in code adapted for use on a digital  
3 computer, said method comprising the steps of:  
4  
5           analyzing the code to determine whether said code  
6           contains instructions causing a macro to be moved to  
7           a global environment;  
8           when the code does not contain instructions causing a  
9           macro to be moved to a global environment, declaring  
10          that no macro virus is present;  
11          when the code contains instructions causing a macro to be  
12          moved to a global environment, flagging said macro;  
13          analyzing the code to determine whether said code  
14          contains instructions causing the flagged macro to be  
15          copied to a local document;  
16          when the code does not contain instructions causing the  
17          flagged macro to be copied to a local document,  
18          declaring that no macro virus is present; and  
19          when the code contains instructions causing the flagged  
20          macro to be copied to a local document, declaring  
21          that said flagged macro contains a macro virus.

22           14. Apparatus for detecting publicly identified and publicly  
23 unidentified macro viruses, said apparatus comprising:  
24  
25           a digital computer having at least one storage device;  
26  
27  
28

1 associated with said digital computer, code containing  
2 computer instructions;  
3  
4 an application program associated with said computer;  
5 a global environment associated with said application  
6 program;  
7  
8 at least one local document generated by said application  
9 program and located within said storage device; and  
10 a detection module coupled to said code, said detection  
11 module analyzing said code and making the  
12 determination that a macro virus is present when said  
13 code contains instructions causing a macro to be  
14 moved to a global environment and said code also  
15 contains instructions causing the same macro to be  
16 copied to a local document.  
17

18 15. The apparatus of claim 14 further comprising a repair  
19 module coupled to the detection module and to the code, said repair  
20 module adapted to delete the code when the detection module  
21 determines that the code contains a macro virus.  
22

23 16. A computer readable medium containing a computer program  
24 for detecting a macro virus in code adapted for use on a digital  
25 computer, said program containing instructions for performing the  
26 steps of:  
27  
28

1 analyzing the code to determine whether said code  
2 contains instructions causing a macro to be moved to  
3 a global environment;  
4  
5 when the code contains instructions causing a macro to be  
6 moved to a global environment, flagging said macro;  
7 analyzing the code to determine whether the code contains  
8 instructions causing the flagged macro to be copied  
9 to a local document; and  
10  
11 when the code contains instructions causing the flagged  
12 macro to be copied to a local document, declaring  
13 that said flagged macro contains a macro virus.  
14  
15  
16  
17  
18  
19  
20  
21  
22  
23  
24  
25  
26  
27  
28